

Εργαστήριο 1 - 2^η Άσκηση - Ανάλυση

Εκφώνηση: Δημιουργείτε εφαρμογή υλοποίησης χρονομέτρου. Θα περιέχει δύο κουμπιά, ένα start και ένα stop. Θα εμφανίζει το χρόνο που μεσολαβεί ανάμεσα στο πάτημα των δύο πλήκτρων.

Υπόδειξη: Θα χρειαστούμε τη βιβλιοθήκη `java.util.*`;

Με την εντολή παίρνουμε την ώρα του συστήματος

`Calendar cal = Calendar.getInstance();`

σε Milliseconds

`double start = cal.getTimeInMillis();`

Άρα για να πάρουμε δευτερόλεπτα πρέπει να διαιρέσουμε με το 1000.

Η εφαρμογή θα μοιάζει ως εξής:



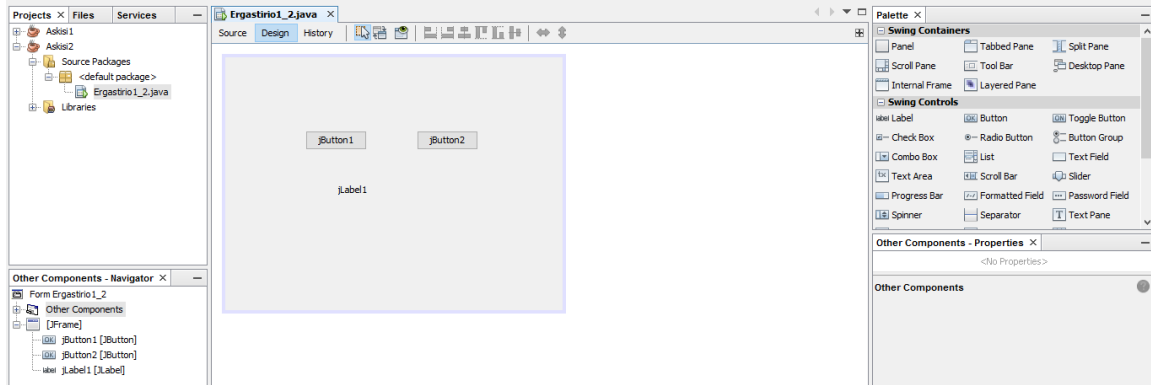
Ανάλυση και εξήγηση λύσης

Βήμα 1: Δημιουργούμε ένα νέο project στο NetBeans με όνομα Askisi2.

Βήμα 2: Δημιουργούμε ένα νέο Java αρχείο τύπου JFrame Form με όνομα Ergastirio1_2.

Βήμα 3: Ορισμός Ιδιοτήτων στο JFrame. Ορίζουμε ως τίτλο του JFrame το «A Simple Stopwatch».

Βήμα 4: Πρόσθεση αντικειμένων στον καμβά και μορφοποίησή τους. Προσθέτουμε δύο JButton και ένα JLabel στον καμβά (JFrame) - Εικόνα 1.

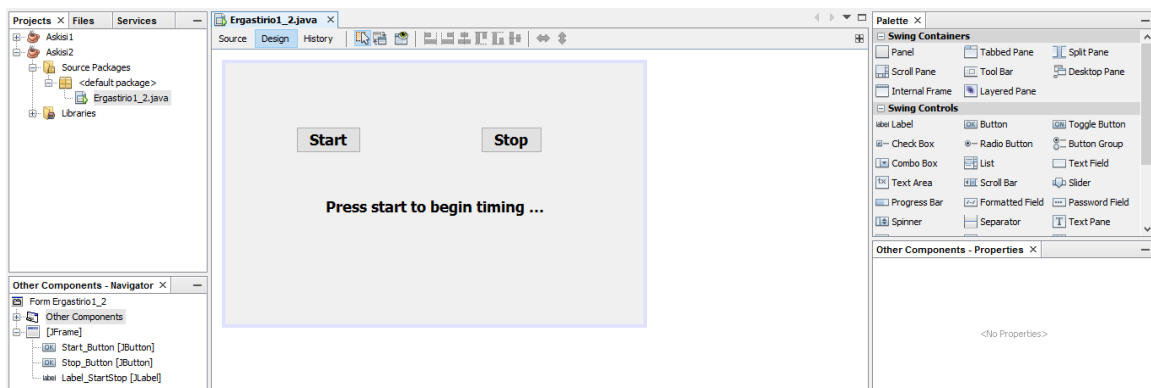


Εικόνα 1: Στο JFrame προστέθηκαν τρία αντικείμενα

Ορίζουμε τις παρακάτω ιδιότητες στα τρία αντικείμενα:

- **JButton1.**
 1. **Text:** «Start»
 2. **Variable name:** Start_Button
 3. **Font:** Bold, 18 Size
- **JButton2.**
 1. **Text:** «Stop»
 2. **Variable name:** Stop_Button
 3. **Font:** Bold, 18 Size
- **JLabel1.**
 1. **Text:** «Press start to begin timing ...»
 2. **Variable name:** Label_StartStop
 3. **Font:** Bold, 18 Size
 4. **Horizontal Alignment:** Center

Η νέα μορφή στον καμβά φαίνεται στην Εικόνα 2.



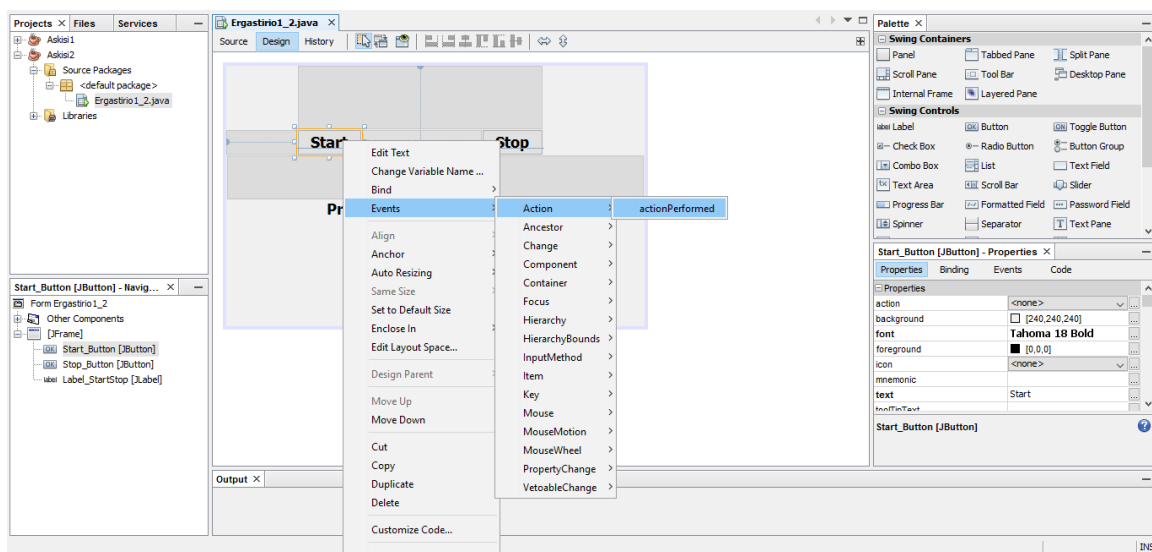
Εικόνα 2: Το JFrame μετά την υλοποίηση του 4ου Βήματος

Βήμα 5: Ορισμός γεγονότων.

- **Start_Button.** Ορίζουμε στο Start_Button το γεγονός Action (Εικόνα 3). Όταν ο χρήστης θα πατάει το Start Button θα πρέπει να γίνουν τα παρακάτω:

1. Να αποθηκεύει τη χρονική στιγμή που το πάτησε.
 2. Να μετατρέπεται η χρονική στιγμή σε milliseconds.
 3. Να ενημερώνεται κατάλληλα ο χρήστης ότι πατήθηκε το Start Button και ξεκίνησε ο χρόνος να μετράει.
- **Stop_Button.** Ορίζουμε στο Start_Button το γεγονός Action. Όταν ο χρήστης θα πατάει το Stop Button θα πρέπει να γίνουν τα παρακάτω:
 1. Να αποθηκεύει τη χρονική στιγμή που το πάτησε.
 2. Να μετατρέπεται η χρονική στιγμή σε milliseconds.
 3. Να βρίσκει τη διαφορά των δύο χρονικών στιγμών.
 4. Να εμφανίζει το αποτέλεσμα της διαφοράς.

Στην Εικόνα 4 φαίνεται ο κώδικας των δύο μεθόδων, με αναλυτικά σχόλια. Έχει προηγηθεί η δήλωση της βιβλιοθήκης Calendar με χρήση της εντολής `import java.util.Calendar;` στη γραμμή 1 του κώδικα.



Εικόνα 3: Δημιουργία Action γεγονότος στο Start Button

```

86 private void Start_ButtonActionPerformed(java.awt.event.ActionEvent evt) {
87     // Παίρνει τη χρονική στιγμή και την αποθηκεύει στο αντικείμενο cal
88     Calendar cal = Calendar.getInstance();
89     // Μετατρέπει τη χρονική στιγμή σε milliseconds - Η starttime έχει δηλωθεί στη
90     // γραμμή 143 ως γενική (global) μεταβλητή
91     starttime = cal.getTimeInMillis();
92     // Εμφανίζει στην ετικέτα ένα μήνυμα
93     Label_StartStop.setText("counting ...");
94 }
95
96 private void Stop_ButtonActionPerformed(java.awt.event.ActionEvent evt) {
97     // Παίρνει τη χρονική στιγμή και την αποθηκεύει στο αντικείμενο cal
98     Calendar cal = Calendar.getInstance();
99     // Μετατρέπει τη χρονική στιγμή σε milliseconds
100    double stoptime = cal.getTimeInMillis();
101    // Στην μεταβλητή finaltime αποθηκεύεται η διαφορά των δύο χρονικών στιγμών σε δευτερόλεπτα
102    double finaltime = (stoptime - starttime)/1000;
103    // Εμφανίζει στην ετικέτα τη διαφορά
104    Label_StartStop.setText("Time passed: "+finaltime);
105 }
106
107 /**...3 lines */
110 public static void main(String args[]) {...31 lines }
141
142 // Δήλωση global μεταβλητής μέσα στην τάξη έχω από τις μεθόδους
143 double starttime = 0;
144 // Variables declaration - do not modify

```

Εικόνα 4: Ο κώδικας του προγράμματος